Multimedia Tools and Applications (2025) 84:2909–2946
https://doi.org/10.1007/s11042-024-20448-9

A systematic review on WebRTC for potential applications and challenges beyond audio video streaming

Haitham Mahmoud[1] · **Raouf Abozariba**[1]

Received: 16 June 2023 / Revised: 24 September 2024 / Accepted: 3 November 2024 /
Published online: 23 November 2024

Abstract
Video conferencing and live streaming are being used in various industries, such as healthcare, gaming, telecommunication, manufacturing and others. As technology progresses, the need for real-time data transmission with minimal latency has increased. Web Real-Time Communication (`WebRTC`) addresses this need effectively. WebRTC is a technology designed to provide real-time communication through web and mobile browsers. Its low latency and P2P communication capabilities make it a convenient technology for secure, efficient communication in real-time applications. This paper reviews the key features of WebRTC, discusses its strengths and weaknesses and investigates a detailed analysis of 83 existing studies. Moreover, It evaluates all use cases that can be adopted by WebRTC by examining their descriptions, problem statements, and research gaps based on literature to date. Finally, It highlights the open research directions for the emerging technologies and enhancements of WebRTC. to identify their potential applications.

Keyword

1 Introduction

The demand for high-quality, low-latency streaming increased sharply due to rise in remote working and the need for real-time collaboration in online gaming, education, and healthcare [1]. WebRTC offers a secure, open-source, and scalable browser-based solution for both individuals and organisations in these fields [2]. As data privacy and security become more critical, the need for secure real-time communication tools is also growing. WebRTC addresses this by providing a platform that protects sensitive user data [3]. Its compatibility with other web technologies makes it easy to integrate into existing systems and applications. The motivation to explore WebRTC beyond video conferencing stems from the increasing need for accessible, secure, cost-effective, and real-time communication capabilities across various applications.

✉ Haitham Mahmoud
haitham.mahmoud@bcu.ac.uk

Raouf Abozariba
raouf.abozariba@bcu.ac.uk

1 Faculty of Computing, Engineering and Built Environment, Steamhouse, Belmont Row, Birmingham B4 7RQ, UK

WebRTC was first released in 2011 and has since become a key component in many applications. Initially, WebRTC was used primarily for video conferencing. With the maturation of the technology, it became apparent that it had the potential to be used for a wide range of real-time communication applications. Among the many applications of this technology are gaming, health, haptic devices, and data collection (such as water sensing). It has been shown that WebRTC can deliver low-latency, real-time communication to enhance the gaming experience for users [4, 5]. This marked a significant milestone in WebRTC's evolution, showcasing its potential beyond just video conferencing. As the healthcare industry's need for secure real-time communication grew, WebRTC emerged as a promising solution. In 2016, Antunes et al. [6] evaluated the feasibility of using WebRTC in telemedicine, demonstrating that it provides a secure platform for real-time communication, including bidirectional file sharing and a whiteboard interface. Today, WebRTC has evolved into a versatile technology in various real-time communication applications, from gaming and education to e-commerce and financial services.

Despite its wide usage, there is a clear research gap in reviewing applications that utilise WebRTC, as highlighted in existing studies. This study aims to systematically review the key applications that use WebRTC as an enabling tool, evaluating them based on criteria such as user experience, network connectivity, server infrastructure, interoperability and security. Moreover. we examined the adaption requirements for supporting non-video and non-audio data, typical attacks on WebRTC, and mitigation techniques. Furthermore, we comprehensively reviewed the existing and potential applications that can be enabled by WebRTC, focusing on problem statements, existing literature, and research gaps for each of them. Finally, we addressed the future research challenges for WebRTC in terms of different emerging technologies. The main contributions of this study are summarised below:

- We conducted a systematic literature review in four phases of identifying relevant studies and examining academic articles related to WebRTC. The method excludes papers describing WebRTC applications without explaining their use and studies written in languages other than English. This review also highlights the need for more scholarly publications.
- We performed an extensive analysis of 83 WebRTC studies, evaluating them based on criteria such as user experience and security requirements. The evaluation highlighted several research gaps.
- Beyond existing WebRTC applications, this study identified additional applications that had not been previously recognised. For each application, we provided a comprehensive analysis of the utilisation of WebRTC as enabling technology, emphasising the problem statement, current literature, research gaps, and the importance of these gaps.
- This study analysed the research gaps in WebRTC applications and reviewed survey studies to highlight open research challenges. This study also examined challenges related to devices and other open issues, offering a roadmap for future research.

The rest of the paper is organised as follows as shown in Fig. 1: Section 2 covers the evolution of WebRTC architecture, its adaptation to non-video data, and a review of the typical attacks and mitigation techniques. Section 3 proposes a systematic framework for analysing WebRTC applications by addressing the key requirements for its applications and presenting the survey methodology. Section 4 discusses the results from the systematic framework, reviews the existing survey papers, examines the advantages and disadvantages, and provides a comprehensive review of each application in terms of the problem statement, literature to date and research gap. Section 5 investigates the open research directions within the WebRTC domain. Finally, Section 6 concludes the work.

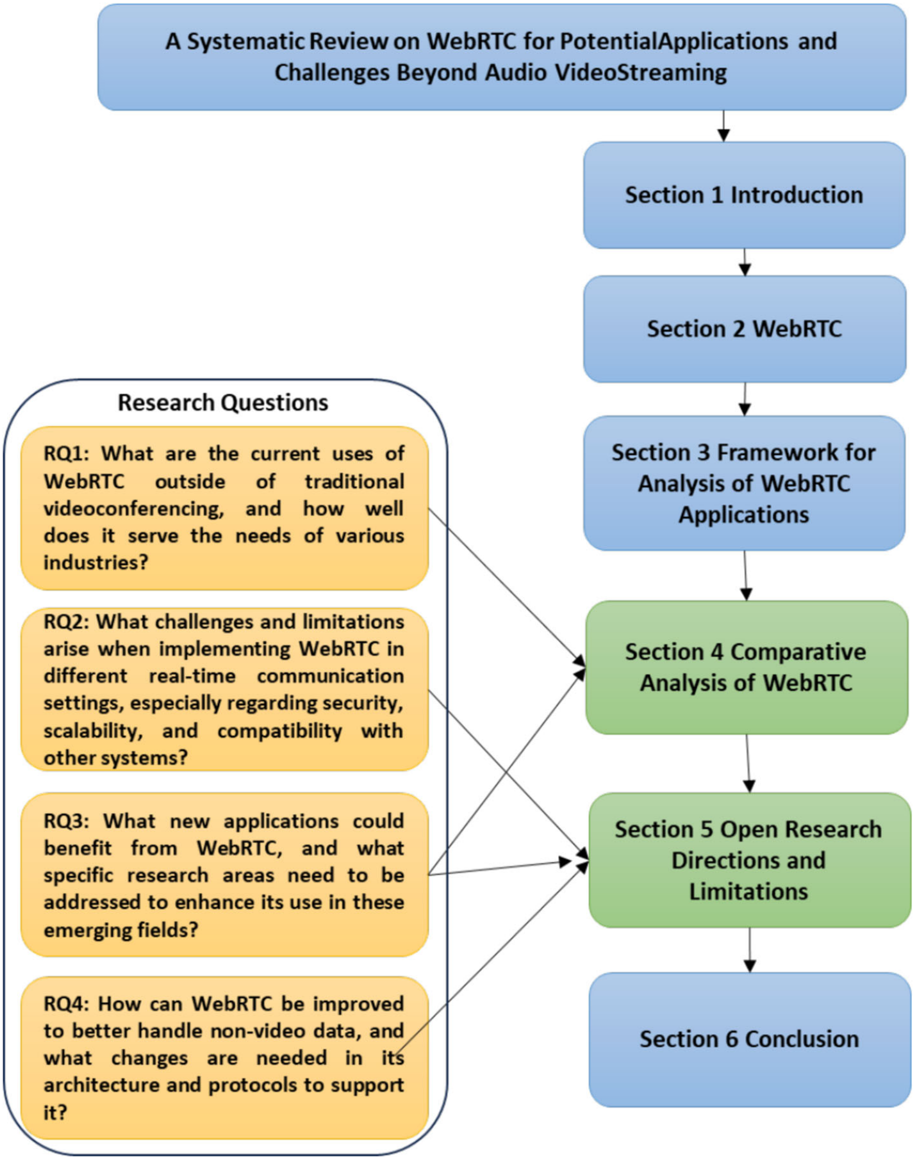


Fig. 1 Structure of the paper

2 WebRTC

2.1 Motivation and research questions

This survey was driven by the increasing relevance of real-time communication technologies in our digitally connected world. WebRTC is an open-source platform, which has become a key technology for enabling real-time communication through web browsers and mobile devices. Despite its success, most existing studies focus on its traditional use in video conferencing, leaving a gap in understanding its broader applications. As industries like healthcare, gaming, and education continue to evolve, there's a pressing need to explore how WebRTC can be adapted to meet the unique demands of these sectors. This survey seeks to fill that gap by thoroughly reviewing the literature, identifying new opportunities for WebRTC, and addressing the challenges that must be overcome to unlock its full potential. The survey is written based on these research questions:

- **RQ1**: What are the current uses of WebRTC outside of traditional video conferencing, and how well does it serve the needs of various industries?
- **RQ2**: What challenges and limitations arise when implementing WebRTC in different real-time communication settings, especially regarding security, scalability, and compatibility with other systems?
- **RQ3**: What new applications could benefit from WebRTC, and what specific research areas need to be addressed to enhance its use in these emerging fields?
- **RQ4**: How can WebRTC be improved to better handle non-video data, and what changes are needed in its architecture and protocols to support this?

2.2 Background

WebRTC allows browsers to communicate peer-to-peer (P2P) in addition to the traditional client-server model. The WebRTC architecture is based on a model similar to the Session Initiation Protocol (`SIP`) Trapezoid (see Fig. 2).

In this model, WebRTC enables web applications downloaded from different web servers to connect both browsers [7]. There are several other technologies available that enable real-time communication, each offering distinct advantages. SIP is commonly used for voice

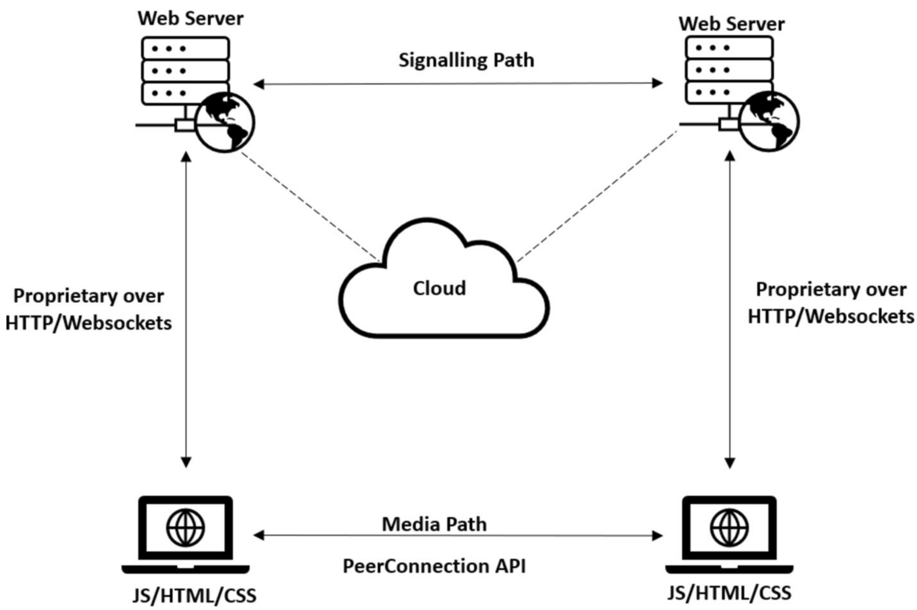


Fig. 2 Typical WebRTC Architecture of two web servers handling signalling between browsers using proprietary protocols over HTTP or WebSockets. After signalling, the browsers establish a direct P2P media path through the WebRTC PeerConnection API, with the cloud representing external services like STUN/TURN servers for secure communication

and video calls, particularly in VoIP systems. RTMP is often used for live video broadcasts, while HTTP Live Streaming (HLS) breaks video into chunks, making it more suitable for large-scale streaming, although with increased latency. Proprietary systems like those used by Zoom and Microsoft Teams provide specialized real-time communication solutions but are generally less adaptable than the open-source WebRTC. Additional protocols such as QUIC and Extensible Messaging and Presence Protocol (XMPP) offer secure communication but are less widely used. Tools like Google Remote Procedure Call (gRPC) support fast, low-latency communication between distributed systems. However, compared to WebRTC, these alternatives often require more server resources and may not offer the same level of peer-to-peer flexibility. Signaling messages are used to set up and end communications. These messages are sent via HTTP or WebSocket protocols to the web servers, which can modify or manage them as needed. WebRTC does not standardise the signalling between browsers and servers since it is considered part of the application [8]. Data is transferred between browsers directly via a PeerConnection without an intermediary server. The two web servers can use a standard signalling protocol such as SIP. Alternatively, a proprietary signalling protocol may be used.

In most WebRTC cases, both browsers run the same web application from the same site. This simplifies the architecture, changing the Trapezoid model into a Triangle (see Fig. 3). The transition from trapezoid `SIP` to triangle architecture involves tools like Network Address Translation Traversal (`NATS`), Interactive Connectivity Establishment (`ICE`), Session Traversal Utilities for NAT (`STUN`), and Traversal Using Relay around NAT (`TURN`) to ensure secure communication [9]. `NATS` provides secure message delivery for real-time communication. ICE is a technique for traversing firewalls and `NATs` to allow direct P2P communication. `STUN` determines a device's public IP and port, while TURN acts as a relay server if direct communication fails. These components work together to ensure secure and reliable communication in WebRTC architecture.

For example, in WebRTC's triangle model, the first user's browser sends a message to the second user's browser using `NATS` to start a video call. `NATS` handles the secure delivery of this message. Once the second browser receives the message, the `ICE` negotiation begins, with each browser sending STUN requests to find their public IP and port and check for NATs. The STUN server helps determine this information. If a direct P2P connection is possible, the video call is established. However, if network restrictions prevent direct communication, one browser acts as a `TURN` server, relaying the video and audio data between the two servers. `TURN` acts as a backup to ensure communication even if NATs block direct connections. Overall, the signalling mechanism, `NAT`, `ICE`, `STUN`, and `TURN`, work together to ensure secure and reliable communication between the two browsers involved in the video call. The details about these components are defined in the next subsection.

2.3 Architecture

The key components of WebRTC are essential for enabling real-time communication between browsers (See Fig. 3). First, the **Media Path API** allows the system to access media streams, such as audio and video, directly from the user's device. This is crucial for applications like video conferencing or voice calls, where capturing and transmitting media is fundamental. **PeerConnection API** sets up a P2P connection between two browsers, facilitating the transfer of media streams and other types of data directly between them. This ensures efficient and low-latency communication, bypassing the need for a server to handle the data exchange. Complementing this is the **RTCDataChannel API**, which provides a secure and efficient

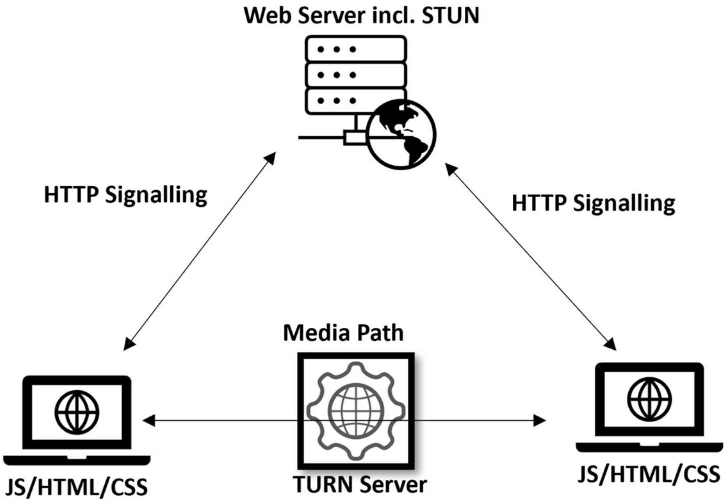


Fig. 3 Secure WebRTC Architecture of two browsers connecting to a web server, including a STUN server, via HTTP signalling. Media communication between the browsers is routed through a TURN server to ensure secure and reliable data transmission, particularly in scenarios where direct P2P connections are not feasible due to network constraints

way to transfer non-media data over the P2P connection, ensuring smooth communication for file sharing or real-time messaging.

WebRTC also uses various **Codecs** for media compression. For video, codecs like VP8, VP9, H.264, and H.265 are employed, while Opus and G.711 handle audio compression, ensuring efficient use of bandwidth while maintaining media quality. To handle **NAT Traversal**, WebRTC uses ICE, allowing devices behind different networks or firewalls to connect directly. **Signaling** is another critical component, used to exchange connection information between devices. It can be implemented using various protocols such as SIP, Jingle, or even a custom protocol. **ICE**: a technique for establishing direct communication between browsers by traversing firewalls and `NATs`. As part of the NAT traversal process, **STUN** helps discover a device's public IP and port behind a NAT, while **TURN** acts as a relay server when direct communication between devices is not possible. **Security** is a top priority in WebRTC, which uses Secure Real-time Transport Protocol (`SRTP`) and Datagram Transport Layer Security (`DTLS`) to encrypt the media and data streams. It also uses ICE to ensure the connection is secure and private.

To use WebRTC for data transfer instead of video, the architecture would need adjustments to support this new functionality [10]. First, the data channel would need to be optimised to efficiently transfer non-video data, potentially requiring the addition of new features or protocols. This would ensure the data transfer is fast and reliable, without data loss during transmission. Another aspect that would need attention is the codecs used by WebRTC. The video codecs may not be suitable for efficient data transfer, so new codecs may need to be developed or modified. This would allow for more efficient compression and decompression of data, improving the overall speed and reliability of data transfer.

Second, the network protocols used by WebRTC may need to be updated or modified to accommodate the transfer of non-video data. This could include changes to how data is transmitted and received to ensure efficient and fast transfer. Security would also be a key concern when transferring non-video data. The security measures for WebRTC video conferencing may need to be adapted or extended to ensure secure data transfer. This would include encryption of data in transit, secure authentication of users, and protection against hacking and other forms of cyberattacks.

Finally, interoperability would be important when using WebRTC for data transfer. To support data transfer between different systems and applications, the interoperability of WebRTC may need to be improved, potentially including the development of new APIs or the modification of existing APIs. This would allow for seamless communication between different systems and applications, making it easier for users to transfer data.

3 Framework for analysis of WebRTC applications

3.1 Key requirements

When implementing WebRTC technology in applications, several key factors should be considered. By addressing these factors, the quality and suitability of a WebRTC application can be effectively evaluated for any specific use case. These factors are as follows:

- User experience is critical, focusing on how easily users can initiate and use the application and the overall quality of audio and video communication.
- Network connectivity is to ensure the application can handle varying network conditions such as low bandwidth or high latency.
- Security is also essential, requiring the application to protect sensitive user data during transmission.
- Server infrastructure must be scalable and reliable to support large volumes of users and traffic.
- Interoperability is another consideration, ensuring the WebRTC application works seamlessly with other systems or communication platforms.
- Real-time communication must be supported to meet the demands of immediate data or media exchange.

3.2 Survey methodology

To thoroughly evaluate the potential of using WebRTC in the literature, we employed a systematic review method developed by Keele et al. [11] to analyse and examine existing literature in this research area (see Fig. 4). This method specifies each stage of our investigation are provided in this section. At the beginning of our research, we conducted a preliminary review of recent literature to better understand the problem and main contributions. Once we determined the feasibility of the research, we divided our systematic review methodology into four stages: selection, identification, screening and refinement, and compilation. These stages allowed for a structured and unbiased collection, evaluation, and synthesis of data pertinent to WebRTC, ensuring the inclusion of high-quality studies. These four stages are described as follows:

3.2.1 Selection phase

In the selection phase, we first chose four scientific databases to extract relevant publications from IEEE Xplore, Springer, Science Direct, and ACM. We used keywords like “WebRTC applications” and “Potential applications for WebRTC” to find relevant material. Moreover,

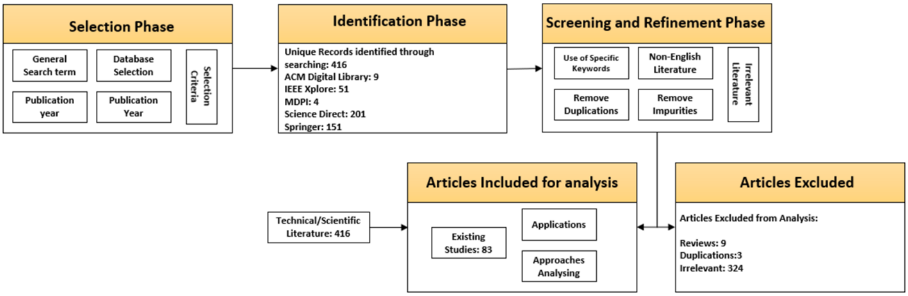


Fig. 4 Survey Methodology of the Systematic Literature review by Keele et al. [11] of four phases; selection phase, Identification phase, screening and refinement phase and compilation of results phase

we used existing conference and journal publication studies to carry out our preliminary study. We utilised the following terms in the advanced research in the databases 'Applications' AND 'WebRTC'.

3.2.2 Identification phase

We reviewed further works on the topic using the snowballing methodology. Within the review, we have found the studies' repetition and duplication of a couple of studies. This led to a deliberate effort to remove all such instances so that items appearing twice were only counted against the publisher's database. We found 83 unique papers across all databases.

3.2.3 Screening and refinement phase

To filter the literature, we performed a screening and refinement phase following the general search conducted in the previous phases. The screening process involved an initial assessment of the titles and abstracts of all studies to determine their relevance to our research focus on WebRTC applications. studies that clearly did not meet the criteria described in Section 3.1. Moreover, studies that focused on unrelated technologies or general communication protocols without direct ties to WebRTC were excluded at this stage.

In the refinement stage, a more in-depth evaluation was conducted. This involved a full-text review of the remaining articles to assess the methodological quality, relevance, and depth of findings. Each article was scrutinised against a set of predefined inclusion and exclusion criteria:

Inclusion Criteria: Studies were included if they specifically addressed WebRTC applications, provided empirical or theoretical evaluations of its use cases, or discussed the potential of WebRTC in real-time communication environments. studies that demonstrated clear experimental setups, validation, or case studies of WebRTC implementations were prioritised.

Exclusion Criteria: Studies were excluded if they were not written in English, lacked a clear methodology, or if WebRTC was only mentioned superficially without substantial analysis or discussion. Duplicate studies, opinion pieces, and publications that did not undergo peer review were also removed. Moreover, we removed articles in languages other than English and papers that did not mention using WebRTC as an application.

Furthermore, studies were assessed for bias, ensuring that findings were not overly influenced by commercial interests or conflicts of interest. The refinement process also involved cross-referencing the findings of the selected studies to identify consistency and gaps, providing a comprehensive view of the current state of WebRTC applications.

3.2.4 Compilation of results phase

As a result of the revisions, 83 unique publications were selected to be included in this study; these papers are classified based on their application. The articles identified each application that can use WebRTC, the literature to date, and the research gap, as discussed in the following Sections 4 and 5. Using forward snowballing, additional outcomes are also extracted from each application scenario.

3.3 Methodology analysis

Figure 5 presents an analysis of the extracted publications based on publication year, type, top 10 subject areas (keywords), and top 10 publishers. The publications per year highlight the growing importance of real-time communication over time. Most contributions are conference papers, indicating quicker, smaller-scale research with room for further development. Key terms such as QoE, real-time communication, P2P, and web applications reflect the relevance of the systematic literature review's focus. Leading publishers, including Springer, Elsevier, MDPI, ACM, and IEEE, emphasize the significance of this research area.

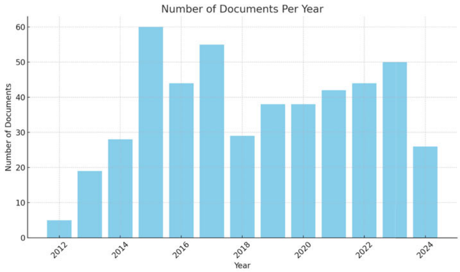
4 Comparative analysis of WebRTC applications

4.1 Results and discussion

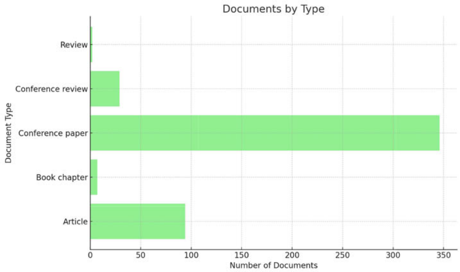
The survey methodology followed in this study involved a thorough examination of 83 research papers on WebRTC, with a focus on the key requirements of user experience, network connectivity, security, server infrastructure, interoperability and real-time performance. The research papers were sourced from multiple reputable databases, including IEEE Xplore, ACM Digital Library, and SpringerLink, spanning. studies were selected based on their relevance, ensuring a comprehensive coverage of WebRTC advancements and challenges. The analysis was based on the key criteria of user experience, network connectivity, security, server infrastructure, interoperability, and real-time performance (see Tables 1, 2, 3). The inclusion criteria for the papers involved studies that explicitly discussed the technical and implementation aspects of WebRTC. Exclusion criteria were set to eliminate papers that focused solely on theoretical or speculative analysis without providing empirical data or prototype implementation. Each paper was rigorously reviewed to extract key insights related to the six criteria, with findings documented in a structured format to facilitate comparison and synthesis.

Most studies in the literature review have focused on enhancing video-conferencing applications and how they can be applied in different industries such as healthcare, gaming, and virtual reality. The primary focus of most studies has been to improve the quality of transmitted data to enhance the user experience, especially given that WebRTC works with multimedia data and any data loss can significantly impact performance.

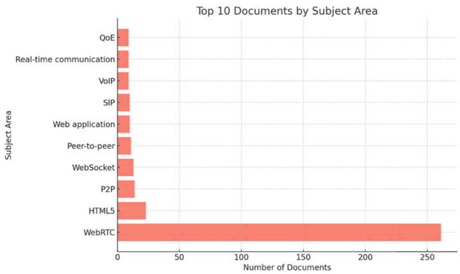
The network connectivity of WebRTC systems has been a key area of focus, emphasising the parameters of reliability and scalability. Research highlights that connectivity issues often arise from NAT traversal and firewall complications, which significantly impact the quality of service (QoS) in real-time applications. Solutions such as STUN, TURN, and ICE protocols have been explored to mitigate these issues but remain suboptimal under high network congestion scenarios [12]. While the transmitted data is of high quality, it must also be stable and reliable to meet the requirements of real-time applications. Scalability



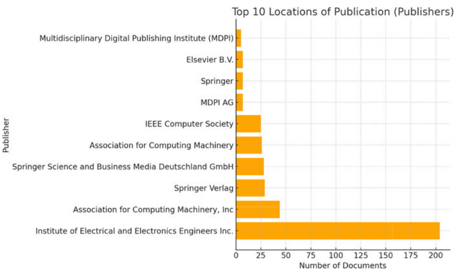
(a) Publications per year



(b) Types of Publications



(c) Top 10 Subject area



(d) Top 10 Publishers

Fig. 5 Data Visualisation of the Extracted papers

remains a challenge in the field, and further research is needed to address this issue. The performance of WebRTC systems has not been fully assessed, which may result in poor real-time performance and high latency when the system cannot accommodate many users.

Concerning security, only a few studies have discussed using cryptography to encrypt transmitted data to maintain its confidentiality [13]. The Secure Real-time Transport Pro-

Table 1 Framework Results from Systematic review on Video and Speech Systems

Ref	Description	QoE/QoS	Network connectivity	Security	Server Infrastructure	Interoper- ability	Real-time
Video and Speech Systems							
[23–26]	Video Streaming in 5G Networks	High	Low Latency	No	Reliable	No	Yes
[14, 27, 28]	browser-based P2P video streaming in resource constrained devices	High	Low BW and Latency	No	Reliable and Scalable	No	Yes
[29]	WebRTC signaling mechanisms and build video conferencing	High	Low latency	No	Reliable	No	Yes
[30]	An implementation of Audio transmission using WebRTC	High	Low BW and Latency	No	Reliable	No	Yes
[31]	Real-Time Analysis of Video Streams on edge	High	Low Latency	No	Reliable and Scalable	Yes	Yes
[32]	Comparing real-time screen-sharing technologies	High	Low Latency	No	Reliable	No	Yes
[33, 34]	performance analysis of client-server applications by SIP and WebRTC.	High	Low Latency	No	Reliable	No	Yes
[35]	Implementing a novel WebRTC signalling mechanism	High	Low BW and Latency	Yes	Reliable	No	Yes
[36]	Investigating the privacy for WebRTC using symmetric encryption algorithm	High	Low Latency	Yes	Reliable and Scalable	No	Yes
[37]	Conceptualising privacy-preserving and maintaining security between peers.	High	Low Latency	Yes	Reliable and Scalable	No	Yes

Table 1 continued

Ref	Description	QoE/QoS	Network connectivity	Security	Server Infrastructure	Interoper- ability	Real-time
Video and Speech Systems							
[38]	Full Reference (FR) analysis of video and audio models using WebRTC	High	Low Latency	No	Reliable	No	Yes
[39]	Synchronised mixing of audio/video streams from multiple peers.	High	Low Latency	No	Reliable	No	Yes
[40]	A server-less framework for edge-based live-video streaming.	High	Low Latency	No	Reliable	Yes	Yes
[41]	A WebRTC-compliant framework for scalability.	High	Low Latency	No	Reliable and Scalable	No	Yes
[42]	Utilising a Kalman filter for congestion control	High	Low Latency	No	Reliable	No	Yes
[43]	Web based video and audio streaming challenges	High	Low Latency	No	Reliable	No	Yes
[44]	Securing web real-time communication using encryption	High	Low Latency	Yes	Reliable	No	Yes
[45]	Videoconferencing security models based on simple access control lists and capabilities.	High	Low Latency	Yes	Reliable	No	Yes
[46]	An adjusting bitrate algorithm for speech applications	High	Low BW and Latency	No	Reliable	No	Yes

Table 2 Framework Results from Systematic review on Streaming Applications, agaming, AR, and VR applications and P2P video and file sharing applcations

Ref	Description	QoE/QoS	Network connectivity	Security	Server Infrastructure	Interoper- ability	Real-time
Streaming applications							
[47]	A load testing WebRTC services	High	Low Latency	No	Reliable	No	Yes
[48]	Introduced WebRTC for the open web platform.	–	–	–	Reliable	Yes	Yes
[49, 50]	Live streaming application	High	Low Latency	No	Reliable	No	Yes
Gaming, AR and VR applications							
[48, 51, 52]	Introduced WebRTC for open web platform.	–	–	–	Reliable	Yes	Yes
[53, 54]	Displaying Videos on computer screens using AR.	High	Low BW and Latency	Yes	Reliable	Yes	Yes
[55, 56]	Interactive Video Conferencing	High	Low BW and Latency	Yes	Reliable	Yes	Yes
[57]	360 degree VR	High	Low BW and Latency	Yes	Reliable	Yes	Yes
[58]	Interactive Educational Platform	High	Low BW and Latency	Yes	Reliable	Yes	Yes
[59, 60]	Utilising WebRTC for game streaming	High	Low BW and Latency	Yes	Reliable	Yes	Yes
[61, 62]	Developed to support multiplayer gaming	High	Low BW and Latency	Yes	Reliable	Yes	Yes
[63]	Cloud gaming using WebRTC and Nvidia GeForce	High	Low bandwidth	Yes	Reliable	Yes	Yes
P2P Video and file sharing applications							
[64–66]	file sharing decentralised system	High	Low BW and Latency	Yes	Reliable	Yes	Yes
[67]	DICOM sharing system	High	Low BW and Latency	Yes	Reliable	Yes	Yes
[68]	Selenium automation for WebRTC testing	High	Low latency	No	Reliable	Yes	Yes
[69]	QUIC-based signaling for WebRTC in impaired networks	High	Low latency	No	Reliable	Yes	Yes

Table 3 Framework Results from Systematic review on Healthcare Systems, Websites, Mobile real-time analyses applications, Censorship Circumventing, Educatoinal Systems, IT applications and Vehicular Systems

Ref	Description	QoE/QoS	Network connectivity	Security	Server Infrastructure	Interoper- ability	Real-time
Healthcare systems							
[70]	Behavioural health clinical care	High	Low BW and Latency	Yes	Reliable	Yes	Yes
[71]	A Medicine file sharing	High	Low Latency	Yes	Reliable	Yes	Yes
[72]	Telepresence wheelchair with 360-degree vision for healthcare applications	High	Low Latency	Yes	Reliable	Yes	Yes
[73]	Medical Image presentation	High	Low Latency	No	Reliable	Yes	Yes
[74–82],	Telehealth consulting System	High	Low Latency	No	Reliable	Yes	Yes
[83]	Interactive Monitoring system using AR	High	Low Latency	No	Reliable	Yes	Yes
[84]	Extended Reality for telemedicine using WebRTC	High	Low latency	Yes	Reliable	Yes	Yes
Websites, Mobile real-time analytics applications and machine learning processing							
[85 ?]	Mobile based analytics system	High	Low BW and Latency	No	Reliable	Yes	Yes
[86–88]	Automated machine learning	High	Low BW and Latency	No	Reliable	Yes	Yes
Censorship circumventing							
[89, 90]	Censorship circumventing	High	Low BW and Latency	No	Reliable	Yes	Yes
Educational Systems							
[91–94]	An online Educational system	High	Low BW and Latency	No	Reliable	Yes	Yes
IoT applications							
[95]	A heterogeneous network for web D2D on the devices	High	Low Latency	No	Reliable and Efficient	Yes	Yes
[96]	An auditory impacted assistive technology	High	Low Latency	No	Reliable	Yes	Yes
[97]	Smart water supply systems using WebRTC	Medium	Low bandwidth	No	Reliable	Yes	Yes
Vehicular Systems							
[12]	Connected and autonomous vehicle discovery	High	Low Latency	No	Reliable	Yes	Yes

tocol (SRTP) is commonly used; however, vulnerabilities still exist, particularly concerning man-in-the-middle attacks and Denial of Service (DoS) threats. Recent advances suggest incorporating more robust cryptographic algorithms, but implementation remains limited in practical applications.

However, the interoperability of WebRTC systems with other technologies has not been widely considered in the studies conducted to date [14]. Interoperability issues particularly arise when integrating WebRTC with legacy systems or non-browser-based applications, hindering seamless communication and data exchange across platforms.

In WebRTC development, the software and tools used in the literature are leveraged to establish real-time peer-to-peer communication, ensuring seamless audio, video, and data transfer. The core WebRTC APIs, such as RTCPeerConnection, RTCDataChannel, and getUserMedia(), are central to capturing media streams and handling peer connections directly within web browsers [14]. RTCPeerConnection is primarily responsible for establishing and maintaining direct connections between peers, and facilitating the exchange of audio, video, and other data streams without the need for an intermediary server. RTCDataChannel allows developers to transfer arbitrary data, including files and game states, over low-latency, bidirectional connections, which is critical for applications beyond traditional video conferencing [15]. Adapter.js is often implemented to resolve cross-browser compatibility issues, providing a JavaScript shim that normalizes WebRTC behaviour across different browsers such as Chrome, Firefox, and Safari, thereby enhancing the reliability of WebRTC applications in production environments [16].

Libraries like PeerJS and SimpleWebRTC simplify the process of peer communication and data transfer [17]. PeerJS abstracts the complexities involved in peer-to-peer networking, providing an easy-to-use API that automates signalling and peer connection management, while SimpleWebRTC builds on top of the core APIs, adding features like room management, video grids, and a higher-level approach to peer discovery and communication. In more complex use cases, media servers like Kurento, Janus, and Jitsi are utilized to manage large-scale, multi-peer communication, enabling features like media transcoding and video conferencing. Kurento Media Server, for instance, allows for real-time processing of multimedia streams, offering advanced capabilities like augmented reality filters and computer vision analysis. Janus, on the other hand, is known for its modularity, supporting multiple plugins that cater to specific needs such as streaming, recording, and SIP gateway functionality. Jitsi, widely recognized for its ease of deployment, is often used in scalable video conferencing solutions due to its robust support for simulcast and SVC (Scalable Video Coding).

For signalling, developers often rely on WebSockets or Socket.IO to manage connection setups and exchange signalling data, such as session descriptions and ICE candidates [18]. WebSockets provide a full-duplex communication channel over a single, long-lived connection, essential for real-time bidirectional data exchange between the client and server, whereas Socket.IO adds additional layers of abstraction, including automatic reconnection, disconnection detection, and multiplexing capabilities. STUN and TURN servers assist with NAT traversal, helping peers discover their public IP addresses and facilitating connections when direct communication is hindered by network restrictions. TURN servers, in particular, relay media between peers in cases where direct peer-to-peer connections fail, albeit with an added latency overhead [19]. Moreover, QUIC is gaining popularity for its reduced latency and network resilience, offering improved performance over traditional TCP connections by incorporating congestion control, forward error correction, and multiplexing within a single connection [20]. SRTP ensures the security of media streams, providing encryption, message authentication, and integrity for the RTP packets, safeguarding media content against eavesdropping and tampering [21].

On the hardware side, WebRTC implementations typically utilize standard device cameras and microphones, with optional integration into specialized hardware setups depending on the application, such as conferencing systems. For professional environments, WebRTC can be integrated into high-end audio-visual systems that include dedicated Pan-Tilt-Zoom (PTZ) cameras, beamforming microphones, and custom Digital Signal Processing (DSP) units to enhance audio and video quality, especially in large conference rooms or live event streaming setups [22].

4.2 Existing review studies

Table 4 demonstrates the existing review papers that touched WebRTC in the literature review. Izima et al. [98] reviewed the machine learning techniques for predicting video quality, including WebRTC applications. Two key findings are extracted from this study: (a) The authors of [99] proposed a method to monitor streaming quality for WebRTC-based audio and visual communication services. They used machine learning models to identify the root causes of video quality problems by analysing various application-layer performance statistics. They used decision trees, random forests, Naive Bayes, SMO, KNN, and bagging models to classify impairments such as video blockiness and audio distortion. They found that decision tree models with high accuracy detecting all three targets. In a similar study, the authors of [100] evaluated WebRTC measurements on a Wi-Fi network and used machine learning models to predict video freezing events. They found that the random forest model had the best performance. These techniques aim to create self-healing systems that can adjust their strategies to improve video quality. Future research could improve these techniques by using multi-dimensional models and incorporating a broader range of audio-visual impairments into evaluations.

There has been a lot of discussion about the potential technologies and challenges of AR in Qiao et al., [101], where a discussion of WebRTC has been made as the communication channel for AR. The authors of Zoran et al. [102] have reviewed the media streaming technology for live streaming events (i.e., concerts). Within that context, WebRTC has been mentioned as one of the potential technologies since it can provide low latency and real-time streaming.

A variety of attacks on WebRTC can compromise security and privacy. MITM attacks, fake STUN/TURN servers, leaks of WebRTC IP addresses, and WebRTC data leaks are common. Media and signalling data are encrypted with DTLS-SRTP and WSS to mitigate the risk of MITM attacks. By using shared secrets or certificates, WebRTC protects against fake STUN/TURN server attacks and authenticates TURN servers using certificate-based authentication. WebRTC also allows users to disable WebRTC or use browser extensions that protect data and IP addresses from leaks and uses STUN servers to determine the public IP address. Users are encouraged to be cautious and take additional steps to protect their privacy and security when using WebRTC despite the current secure architecture, which provides several security features. Moreover, users can utilise VPNs to enhance their privacy and security further by disabling WebRTC, using browser extensions that block IP and data leaks, and using browser extensions that block IP leaks.

No review paper explores WebRTC and addresses its potential applications and significance to the best of our knowledge. Therefore, We examined the technology's benefits, shortcomings, and potential applications to learn about its significance in this paper.

Table 4 Existing Review studies about WebRTC

Ref	Description	App.	studies	Pros	Cons
Izma et. al., 2021 [98]	A review of ML techniques for video quality prediction	Video conferencing	2	Best ML for video quality prediction	did not focus on WebRTC
Qiao et al., 2019 [101]	Review advantages challenges of AR including WebRTC	AR	1	Presented the motivation	did not focus on WebRTC
Zoran et. al., 2022 [102]	reviews the media streaming technologies	Streaming	3	reviews WebRTC as potential approach	did not focus on WebRTC

4.3 Advantages and challenges

4.3.1 Advantages

There are several advantages brought about by WebRTC that can be summarised as follows:

- The technology is an open-source system that can be easily integrated into other applications, contributing to its accessibility to various industries and organisations.
- Because WebRTC provides low-latency communication, it is ideal for real-time applications such as video conferencing and live streaming, enabling users to enjoy high-quality and responsive communication experiences.
- It enables P2P communication, which reduces server dependency and enhances scalability, making it an attractive solution for large-scale applications.
- Secure communication is ensured by WebRTC's integrated security features, such as encryption and certificate management.

4.3.2 Challenges

In contrast, WebRTC has some challenges such as system complexity, interoperability, security risks, firewall constraints, ambitious requirements, and standardisation. These challenges can be summarised as follows:

- The complexity of WebRTC's architecture and the high level of technical expertise required to implement it may make it difficult for users without technical expertise to utilise it.
- The majority of browsers currently support WebRTC, including Chrome, Firefox, and Opera. Users who rely on Internet Explorer or Safari may be unable to use it due to this limitation. It cannot be used by users using different browsers or platforms due to the requirement that both the sender and the receiver use compatible browsers.
- Communication via WebRTC relies on User Datagram Protocol (`UDP`), which may lead to communication issues with some networks and firewalls that block or restrict UDP packet transmissions.
- There are potential security risks associated with using WebRTC, despite the technology's built-in security features, such as encryption and certificate management. The privacy and security of WebRTC users may be compromised due to WebRTC leaks, for example.
- In low-bandwidth or high-latency network conditions, WebRTC may be resource-intensive and may not perform well. This can affect the quality of communication.

4.4 WebRTC enabled applications

In this section, we undertake a thorough examination of both existing and potential applications facilitated by WebRTC technology (See Table 5). Our focus is to analyse each application in terms of its description, problem statement, literature to date, and the gaps that remain unexplored. This review aims to highlight the broad potential of WebRTC across various sectors while identifying key areas where further research and development are required to fully realize its capabilities.

4.4.1 Voice/video chatting and conferencing

Description WebRTC is commonly used for voice and video chatting and conferencing, allowing teams to hold real-time virtual meetings with audio and video streams, under low

Table 5 Summary of WebRTC Enabled Applications

Application	Description	Problem Statement	Literature to Date	Research Gap
Video Conferencing	Real-time video and audio communication between users.	Ensuring high-quality, low-latency communication across devices.	Extensive research on enhancing video and audio quality.	Interoperability with legacy systems and improving QoS.
Gaming Platforms	Enables real-time communication and streaming in online games.	Reducing latency for a seamless gaming experience.	Studies focus on performance and latency improvements.	Enhancing haptic feedback and standardizing data transmission.
Telemedicine	Facilitates remote patient consultations and monitoring.	Secure and reliable data transmission for healthcare applications.	Research on video consultations and remote monitoring tools.	Integration with healthcare systems and improving security.
Live Streaming	Real-time streaming of events, classes, and performances.	Delivering low-latency, high-quality streams to large audiences.	Demonstrated suitability for various live events.	Optimizing for different network conditions and privacy concerns.
P2P File Sharing	Direct file transfer between users without intermediaries.	Ensuring fast, secure, and reliable data transfers.	Proof-of-concept studies on secure file sharing.	Scalability and compatibility with existing file-sharing systems.
Virtual & Augmented Reality (VR/AR)	Enables immersive experiences through real-time interaction and updates.	Low-latency communication and high bandwidth for VR/AR applications.	Initial studies on VR conferencing and interactive AR tools.	Developing standards for VR/AR data transmission.

Table 5 continued

Application	Description	Problem Statement	Literature to Date	Research Gap
Online Marketplaces	Enhances buyer-seller interactions through real-time communication.	Facilitating secure and efficient transactions in real-time.	Few initiatives, with focus on enhancing user experience.	Scaling WebRTC for large platforms and ensuring transaction security.
Educational Platforms	Supports live lectures, group discussions, and virtual labs.	Providing low-latency, high-quality educational experiences online.	Explored in online classrooms and interactive learning tools.	Further development of educational tools and improving scalability.
Weather Sensing	Collects and communicates real-time weather data in urban environments.	high accuracy and timely weather data collection and analysis.	Limited research on urban weather data collection using WebRTC.	Enhancing data accuracy and integrating with existing weather systems.
Censorship Circumvention	Bypasses government restrictions on information access.	Secure, real-time communication without detection.	Some studies on using WebRTC for secure, private communication.	Addressing vulnerabilities and improving scalability.
Machine Learning Processing	Facilitates decentralized and real-time data processing for machine learning.	Ensuring privacy and efficiency in data handling for ML tasks.	Early-stage research on integrating WebRTC with ML processes.	Improving scalability, standardization, and privacy measures.
Haptic - Tactile Internet	Transmits touch sensations over the internet for interactive experiences.	Low-latency, high accuracy transmission of haptic feedback.	Initial studies on teleoperation and medical applications.	Developing advanced hardware and standardizing haptic communication.

latency and high-quality communication. Platforms like Google Meet and Zoom are examples of WebRTC-powered video conferencing solutions. Another use is voice chat, where WebRTC supports applications such as Discord, often used in gaming or social networking, allowing users to engage in seamless voice communication.

Problem statement WebRTC's implementation for voice and video communication focuses on enabling real-time connections between two or more users via a web browser or a mobile app, without the need for additional software or plugins. Key challenges include ensuring device and browser compatibility, achieving optimal network performance, and maintaining secure communication channels. In addition, the basic functionalities of video conferencing should be augmented with vertical tools to enable more access, control and independent customization by users.

Literature to date WebRTC literature has focused primarily on voice and video conferencing, including system enhancements, performance optimisations, and user experiences. Researchers introduced new techniques for managing network congestion [42], processing multimedia in real-time, and P2P communication to improve the performance and scalability of WebRTC systems [31, 36]. Authentication and encryption mechanisms have been integrated into WebRTC systems to enhance security, and privacy [35–37]. Further research has been conducted to improve WebRTC user experiences, including ease of use and interface design. The research community also explored whether WebRTC can be integrated with existing communication systems, such as telephones, to improve interoperability and expand its application scope. WebRTC has also been studied in emerging applications, including virtual reality and the Internet of Things (`IoT`), to provide new dimensions for real-time communication. For a deeper explanation of such potential applications, we separated the IoT applications from this one.

Research gap Aforementioned the literature, the development of WebRTC for voice and video chatting and conference is still in its early stages. Several research gaps need to be addressed. One major challenge is how to interoperate with legacy communication systems such as telephones; another issue is managing network congestion and reducing latency, so users have a positive experience; meanwhile, protecting user data from man-in-the-middle attacks remains an important concern [103, 104]. Additionally, current technology does not have reliable methods to ensure Quality of Service (QoS) for consistent and high-quality audio and video experiences [24]. Multi-view (MV) learning that allows for improving generalization efficiency by learning from multiple viewpoints is one of the future optimizing techniques being explored by the research community [105]. Optimizing Deep neural networks (DNN) parameters such as the number of layers and nodes, which can be leveraged to estimate QoE and QoS of WebRTC streaming remains an open research question [24].

4.4.2 Gaming platforms

Description WebRTC technology is used in the gaming industry to enable real-time communication within web browsers. Without dedicated servers, P2P communication between players provides low latency and high-quality audio and video streams. The result is a more immersive and interactive gaming experience. Game developers can use WebRTC technology to enhance player interaction, including real-time voice and video communication, live gameplay streaming, and transmitting haptic information. WebRTC technology has already been implemented on some gaming platforms such as Twitch, with many more exploring its potential in this sector[106, 107].

Problem statement To support gaming applications effectively, WebRTC technology must meet several requirements. A seamless gaming experience requires low latency in real-time communication. User experience can be negatively affected by latency or the delay between sending and receiving information between gamers. As a result, it is essential to minimise latency when large volumes of data are exchanged. Users must be able to enjoy an interactive and engaging gaming experience with WebRTC as an enabler by ensuring that audio and video are transmitted with high quality. To ensure the smooth and effective operation of gaming platforms, WebRTC must also be able to handle high data volumes and bandwidth requirements.

Literature to date Several studies have been conducted to assess the use of WebRTC in online gaming, focusing on topics such as performance, latency, user experience, and security. For example, study by Sabesan Muralikrishnan found that WebRTC provided low latency and high performance in online gaming compared to traditional gaming solutions [59, 60]. Another study by the National Institute of Standards and Technology (NIST) found that WebRTC was suitable for low-latency multiplayer gaming, providing high-quality performance even in network-constrained environments [61, 62].

Research gap Several research gaps exist when it comes to WebRTC use in gaming. Gaming data must be transmitted with low latency, network congestion must be efficiently managed to minimise lag, haptic hardware and software must be developed to create a more immersive gaming experience, and gaming data must be transmitted in a standard manner to ensure compatibility across all platforms and devices. Despite some initiatives exploring this application, further improvements are needed for reliable streaming with high quality. Leveraging network traffic data and arrangements with service providers, WebRTC can further reduce latency by utilizing application-level metrics, particularly between gamers separated by large physical distances [108].

4.4.3 Streaming platforms

Description Live streaming is also possible using WebRTC, such as concerts, classes, and events. Using the technology, developers can create low-latency, high-quality streaming solutions. YouTube Live is widely recognised as a live streaming service that utilises WebRTC. In addition to a seamless streaming experience, WebRTC allows users to view online broadcasts of live events with minimal delay. Remote audiences can participate in real-time at virtual events, facilitated by this feature.

Problem statement Real-time streaming imposes heavy load ensuring high quality video communication on the network. As the transmitter has to encode video sequences in real-time by deciding the pacing rate adjusting Congestion Control and Adaptive Bit rate considering the network condition. In addition, as video streaming consumes a significant portion of bandwidth, there is a need to allocate bandwidth more efficiently to improve overall user experience. WebRTC is crucial for live streaming applications due to its low latency, (`P2P`) network capabilities, and built-in security features. Low latency ensures seamless, near-instant communication for a high-quality user experience. P2P network reduces server burden and cost, allowing for a scalable solution. Built-in security protects live streaming data from unauthorised access or tampering.

Literature to date WebRTC enables users to stream live events smoothly and with little to no delay, with little to no delay between the live event and the broadcast [49, 50, 109]. Additionally, WebRTC's P2P architecture reduces the load on server infrastructure, resulting in cost savings for streaming services [47, 48]. It has been demonstrated that WebRTC is suitable for streaming various applications, including live sports events, concerts, and other large-scale gatherings. WebRTC has been demonstrated to be effective in providing high-quality, low-latency streaming to many users, despite less-than-ideal network conditions.

Research gap Although WebRTC for streaming can potentially improve video and audio quality and reliability, research gaps remain. In addition to addressing potential security and privacy concerns, one gap is finding effective solutions for WebRTC streaming with high bandwidth and low latency. In addition, WebRTC needs to be optimised for different network conditions and improved to be compatible with existing streaming platforms and technologies.

4.4.4 P2P video and file sharing platforms

Description File sharing is popular tool for web real-time file transfer applications. The common methodology of implementing file-sharing is uploading contents to a file server by a sender, then receivers can download the contents after passing access control and authentication mechanisms [110]. WebRTC enables efficient, secure, and scalable P2P file sharing with improved speed, reliability, and robust security features like end-to-end encryption, without the need for a middle server. When a sender initiates a transfer process, the application generates a link recipients can use to download the file over the WebRTC data channel. This makes it a valuable technology for developers creating file-sharing apps, as demonstrated by Facebook Messenger's P2P file-sharing feature.

Problem statement WebRTC uses four types of servers to establish P2P connection including web hosting, signalling servers, STUN server, and Traversal Using Relays around NAT (TURN) server. WebRTC uses the TURN server in case the STUN server fails. However, the study shows in [111] that 92% connections are established on STUN while 8% connections are established using TURN. Therefore, webRTC has to cope with firewall and NAT devices to establish a real-time P2P connection between two browsers. Efficient and fast data transfer, robust security, and scalability are crucial for WebRTC in P2P file sharing. WebRTC enables direct communication between browsers for faster transfer speeds, offers end-to-end encryption for secure transfers, and has a decentralised architecture for scalability.

Literature to date In some existing studies, WebRTC has been examined as a potential tool for P2P file sharing, including faster transfer speeds, increased reliability, and increased security through end-to-end encryption [64, 65]. A proof-of-concept of developing a DICOM file exchange using WebRTC was introduced in [67]. In addition to highlighting some challenges, these studies have highlighted the need for improved scalability and compatibility with the existing file-sharing protocols [66]. The security and reliability aspects of shared data through a combination of blockchain and WebRTC for decentralized file control was investigated in [112].

Research gap Optimising WebRTC for file sharing, especially regarding speed and reliability, requires further research. As well as security and privacy concerns, WebRTC must be compatible with various file-sharing devices and platforms to prove its potential. Blockchain integration with WebRTC requires further investigation in terms of relay node selection methods and compensation mechanisms to improve the reliability and availability of relays in the future.

4.4.5 Virtual reality and augmented reality platforms

Description WebRTC enhances Extended Reality (XR) app experience with low-latency real-time communication and P2P capabilities. It enables Virtual Reality (VR) users to interact with each other in near real-time and AR users to receive real-time updates, resulting in a more immersive experience. It also provides a scalable solution, reducing the burden on centralised servers.

Overall, WebRTC improves the user experience in VR/AR applications.

Problem statement To achieve low latency in AR and VR applications is challenging as the user experience is influenced if the delay is more than 200ms. The high-end supercomputers and clusters are needed for immersive and virtual reality environments. There is still a need to deal with timing dependencies for distributed applications running on these clusters while implementing virtual reality environments.. WebRTC meets the essential requirements for VR and AR applications, with low latency communication and P2P capabilities for scalability. The technology ensures real-time interactions and updates for an immersive experience, and its built-in security features protect sensitive information from unauthorised access.

Literature to date It appears that the key idea is to apply the concept of augmented reality to video conferencing in general, [55, 56]. There are many ways to engage and interact with users in 360-degree VR [57]. There has been a growing interest in the potential of fully immersive and virtual reality of 3D, and WebRTC has been mentioned as one of the techniques used in this study [52]. As a result of the application of augmented reality in healthcare, telehealth experiences have been enhanced [83]. This is another application considered for online education to provide interaction using virtual reality [58]. WebRTC was also proposed to update the position of the virtual camera to receive the Free Viewpoint Video (FVV) live view encoded as a video. In this context, a WebRTC server is placed between a remote user and a view renderer in the local user to enable a WebRTC connection [113].

Research gap VR and AR applications require WebRTC, which has several requirements. These include low latency for real-time interaction, high bandwidth for transmitting large volumes of VR/AR data, reliable data transmission that ensures a smooth user experience, and compatibility across different VR/AR platforms and devices. Standards and protocols for VR/AR data transmission must also be developed to ensure seamless communication between different devices and applications. In addition, adapting the deployment and the WebRTC Server to handle several simultaneous remote users receiving the same FVV Live transmission remains an open research problem.

4.4.6 Online marketplaces

Description The online marketplace connects buyers and sellers online, creating a new relationship between companies and customers giving a personalized shopping experience. Through real-time communication, WebRTC can enhance online marketplace users' experience. WebRTC enables real-time voice and video calls between sellers and buyers in online marketplaces. As a result, buyers can ask questions and receive live demonstrations from sellers, which allows them to make more informed purchases. Customers and customer service representatives can also communicate in real-time with WebRTC. As another use-case of how WebRTC can be used in online marketplaces, buyers and sellers can collaborate in real-time during the transaction process. WebRTC can, for example, be used by buyers and sellers to digitally sign contracts, which will reduce the time and effort involved in completing the transaction.

Problem statement More than 4 billion users are connected today to internet where 85% of these are online and 93% of these are connected through mobile devices. E-commerce is growing exponentially, especially with the increase in digital advertisement by companies like Google Ads, Display Ads, Youtube Ads, and Gmail Ads [114]. To manage contents on online marketplaces, existing approaches relied on cloud computing and content distribution networks which require large monetary investment and operation costs. To offload some of the distribution costs onto end users, more recent solutions used client-side software or web browser plug-ins but studies reported poor user incentives, which limited their implementation in the industry. WebRTC promises to provide real-time content sharing among companies and customers, end-to-end encryption, and a convenient user experience, enhancing online marketplaces.

Literature to date Several websites and articles have mentioned the idea of utilizing WebRTC in the online marketplace model [115–117], but the concept has not been extensively explored in the literature. For example, [115] proposed the use of WebRTC to enable content distribution using online marketplace scenarios in two modes: (i) client-to-coordinator and (ii) client-to-client. the results show the proposed solution reduces the 95th percentile bandwidth due to image content at the operator by over 75%.

By reducing the time and effort required for buyers and sellers to exchange information, these initiatives demonstrated that WebRTC could enhance the user experience. Consequently, further research is required to evaluate the significance of this application.

Research gap To improve the effectiveness of WebRTC and overall marketplace security, additional research is needed to address its scalability, integration, user experience, and security in online marketplaces. The objective is to ensure the technology that supports the growing demands and enhances the user experience.

4.4.7 Healthcare monitoring systems

Description WebRTC enables virtual healthcare services like virtual consultations, remote patient monitoring and telehealth education. It allows doctors to diagnose patients and provide treatment recommendations through real-time virtual consultations. Moreover, Healthcare can benefit from WebRTC, making care delivery more efficient and improving efficiency in the system. For example, real-time medical information can be exchanged between doctors and patients using telemedicine. As a result, patients may receive better care and travel to medical appointments may be reduced. Furthermore, WebRTC allows healthcare providers to communicate and coordinate more efficiently and effectively, improving patient care. By securely sharing medical information and images, healthcare providers can collaborate more effectively and reduce the possibility of medical mistakes. Additionally, WebRTC can also improve the delivery of remote healthcare services, such as remote patient monitoring and telerehabilitation. As a result, individuals living in rural or remote areas can have easier access to healthcare services and improved quality of care for those unable to travel to a healthcare facility.

Problem statement To ensure the success of WebRTC in telemedicine, reliable and secure data transmission, seamless integration with healthcare systems, standardised regulation, and sufficient connectivity and device compatibility are crucial. Healthcare providers must address data security issues with encryption, secure servers, and firewalls while ensuring compatibility with existing healthcare systems. Standardisation and regulation are necessary to guarantee the quality and reliability of telemedicine services, and adequate connectivity and device compatibility are essential to reach patients in need.

Literature to date Telehealth video monitoring has been explored in a few references [74–79, 81]. The system should be able to connect a doctor with another doctor or patient remotely for diagnosing and follow-ups. Similarly, another contextual health information system from the connected medical sensors is developed. This paper reviewed the characteristics, context and use cases of telehealth applications on the web [80]. An extension to this idea, Augmented reality has been applied to provide more interactive features within the telehealth video conference [83]. This can enable the patients to benefit from medical assistance by calling remote medical support [82].

Research gap Research is still needed to unlock the potential of WebRTC in healthcare. The focus is on the impact of technology on patient outcomes, security and privacy, integration with healthcare systems, and cost-effectiveness. More research is required to determine how WebRTC can improve patient outcomes, secure P2P communication, integrate seamlessly with healthcare systems, and analyse its long-term financial impact.

4.4.8 Websites and mobile real-time analytics

Description With WebRTC, data collection and analysis can be performed in real-time for mobile devices and websites. Users benefit from a seamless user experience, and businesses gain valuable insights from real-time data analytics. Similarly, WebRTC can enable crowdsensing. A crowdsensing technology gathers real-time data on many users through their mobile phones. Data can be collected from mobile devices and transmitted to a central server using WebRTC, a real-time communication technology. By combining data from multiple sources, crowdsensing applications can provide a wide range of real-time insights into various aspects of the environment, including traffic, air quality, and weather.

Problem statement The website or mobile app must communicate with their end users in real-time for real-time analytics. Data can be collected in real-time using WebRTC technology so that companies can get immediate feedback and insights into their users' behaviour. To facilitate real-time communication, WebRTC eliminates the need for users to download additional software or plugins and provides a seamless user experience. As a result, websites and mobile apps have become more engaging and user-friendly.

Literature to date In real-time analytics for websites and mobile devices, WebRTC technology has been adopted since it allows real-time communication within the web browser [85]. Another application develops a mobile-based over-the-cloud solution that takes advantage of the many features of the YOLO algorithm [118]. Studies have been conducted on WebRTC for real-time analytics on websites and mobile devices. WebRTC has also been shown to enhance the user experience by providing real-time support and interaction, which leads to higher customer satisfaction and engagement. Other studies have demonstrated that WebRTC can enhance online marketplaces' functionality, allowing buyers and sellers to communicate and collaborate in real-time.

Research gap Additionally, there is still a lot of research to be done to determine whether WebRTC is reliable and scalable for large-scale data collection, whether data security and privacy measures can be improved, and how WebRTC can be integrated with existing analytics platforms. Moreover, WebRTC-based analytics need to be visualised and communicated most effectively.

4.4.9 Censorship circumventing

Description Censorship circumvention refers to the efforts to bypass or overcome government restrictions or censorship of information or communication technologies. WebRTC technology can be used as a tool to help circumvent censorship in countries where the free flow of information and open communication is restricted. WebRTC provides secure, real-time communication directly from browsers, making monitoring easy for censorship. It offers end-to-end encryption for added security and privacy, making it useful for journalists, activists, and others in censored countries. However, it may still be vulnerable to interception or blocking, and more research is needed to improve its security for censorship circumvention.

Problem statement In censored countries, WebRTC can meet three key needs: privacy, real-time communication, and ease of use. A user-friendly solution for fast and effective communication is offered, which includes end-to-end encryption to ensure secure communications, real-time communication directly from a web browser, and end-to-end encryption.

Literature to date The use of WebRTC for circumventing censorship has been explored in various studies to date. Several studies have explored how individuals and organisations perceive and use WebRTC for censorship circumvention, and their challenges in accessing and using it [89, 90]. In these studies, it was demonstrated that WebRTC does provide a user-friendly solution for circumventing censorship. However, there are still challenges to be overcome, including technical expertise and compatibility with existing circumvention tools.

Research gap There's a need for research on WebRTC scalability, security, user experience and compatibility for censorship circumvention. To ensure WebRTC can support increased demands, address potential vulnerabilities, be accessible and user-friendly, and integrate effectively with existing censorship circumvention tools.

4.4.10 Machine learning and deep learning processing

Description With WebRTC, data can be decentralised and distributed across devices and models, enabling real-time communication and collaboration. As a result, machine learning tasks can be performed more quickly and efficiently, while data transfer and storage costs can also be reduced. Furthermore, WebRTC can allow machine learning to process data without transferring it to a central server, preserving privacy. Various fields, including predictive maintenance, image and speech recognition, and autonomous systems, can benefit from this application of WebRTC.

Problem statement It is imperative to ensure the privacy and security of sensitive data when using WebRTC for machine learning, to manage large amounts of data, and to achieve real-time performance. Additionally, WebRTC devices and machine learning models must be interoperable and standard. A simulated eavesdropping attack on WebRTC was presented in this paper [119]. The privacy concerns within the system must also be considered in addition to the security risk.

Literature to date There has been a growing interest in WebRTC technology's benefits for automating machine-learning processes in real-time. Several studies have been conducted in the past few years to explore the potential of WebRTC in this area [86, 87]. Machine learning algorithms benefit from WebRTC's ability to transfer and process data in real-time, enhancing their speed and accuracy. An analysis of WebRTC's potential for distributed machine learning processing can be found in [88]. Using WebRTC for distributed machine learning is an ideal solution due to its ability to transfer data efficiently between multiple nodes in a network. In this manner, machine learning algorithms can be trained and executed in parallel, resulting in an increase in accuracy and a reduction in processing time.

Research gap A relatively new area of research utilising WebRTC for machine learning is WebRTC-based machine learning. There are a limited number of studies available on this topic. There are several research gaps in the use of WebRTC for machine learning; WebRTC cannot handle (1) Large amounts of data due to its limited scalability; (2) Standardisation of WebRTC implementation for machine learning is lacking; (3) WebRTC has not yet been studied for its performance and efficiency with machine learning; and (4) security and privacy should be considered.

4.4.11 Haptic - tactile internet

Description Haptic technology uses touch sensations and forces feedback to enhance users' experience with digital content and devices. This type of technology can be used to improve navigation through websites or provide more immersive gaming experiences. The haptic internet is a technology that can transmit touch sensations over the Internet. WebRTC enables support for haptic technologies, which allows for more interactive and immersive digital experiences. These benefits may be valuable in gaming and telemedicine contexts, where enhanced experiences could have positive effects.

Problem statement To fully realise the potential of WebRTC for haptic applications, it is necessary to provide low latency, high bandwidth, improved hardware/software, and standardisation. A real-time, seamless haptic transmission is essential to fully exploit the potential of WebRTC for haptic applications. To ensure immediate response and feedback, a low latency connection and high bandwidth are required to accommodate the large volume of haptic data. Furthermore, hardware and software technology advancements are necessary to ensure high accuracy and precise transmission of haptic information. Establishing industry standards to ensure interoperability between different haptic devices and applications is also imperative.

Literature to date Some studies have explored the use of WebRTC for haptic communication in teleoperation, telemedicine, and other applications. Kenm et al. (2021) [120] investigated the use of WebRTC for haptic communication in real-time teleoperation systems, in which remote users can control a robot or other equipment in real-time. Teleoperation applications can benefit from WebRTC's low latency and high reliability when transmitting haptic information. Another study by Kurillo et al. (2016) [121] explored the use of WebRTC to facilitate remote diagnosis and treatment of patients through real-time haptic communication. This study found that WebRTC provided an effective way of transmitting haptic information in real-time, enabling medical professionals to diagnose and treat patients remotely in high accuracy. In this study, WebRTC was demonstrated to have the potential to provide haptic feedback and communication in real-time applications, and further research is needed to improve and refine this technology.

Research gap A full understanding of the potential of WebRTC for haptic applications will require ongoing research. There are four main areas of focus: low latency transmission, optimal bandwidth utilisation, development of advanced haptic hardware and software, and the standardisation of haptic information transmission. There is a need for research to minimise latency, reduce bandwidth for large haptic data, improve haptic hardware and software, and develop widely adopted haptic communication standards.

4.4.12 Weather sensing in urban environments

Description In an urban environment, WebRTC can collect and communicate weather data in real-time. Weather forecasting accuracy can be improved by gaining insight into the current weather. For instance, a central database can be connected to weather sensors throughout a city using WebRTC. In addition to collecting data about temperature, humidity, and wind speed, these sensors can also measure other important variables related to the weather. WebRTC can transfer data from the weather station to the central database for analysis and display in real-time.

Problem statement Weather sensing has various challenges in an urban environment when using webRTC for data collection from various points. These include a lack of high accuracy weather data, effective methods of processing and analysing large amounts of data, and seamless integration with existing weather monitoring systems. These challenges must be addressed to obtain high accuracy and timely information about local weather conditions and formulate an interoperable and comprehensive system for weather sensing.

Literature to date A few amount of literature has been published on using WebRTC for weather sensing in urban environments. It has been demonstrated, however, that WebRTC technology has the potential to improve forecast accuracy and provide real-time weather information. The use of WebRTC in an urban environment can be observed by the collection and transmission of weather data from multiple devices and sensors [122]. UAVs were also used in some studies for data collection [123, 124]. As a result, it is possible to monitor and analyse weather patterns and conditions in real-time, allowing for high accuracy forecasts and early warnings.

Research gap Several gaps must be addressed to fully leverage WebRTC for weather sensing in an urban environment: (1) WebRTC data transmission must be improved for accuracy and reliability; (2) need to develop effective methods for processing and analysing large quantities of weather data collected from multiple sources, (3) review the integration of WebRTC technology with existing weather monitoring systems and networks to provide weather sensing systems that are seamless and interoperable. As cited in [123], more research is needed to improve the accuracy and reliability of collected weather data.

4.4.13 Educational platforms

Description Students and teachers can communicate in real-time online using WebRTC in educational settings, enabling immersive and interactive learning. As a result of its low latency and high-quality audio and video capabilities, it allows for live lectures, group discussions, and remote collaboration, thus improving access to education and lowering geographical barriers. Using WebRTC, educational tools and applications such as virtual labs and simulations can be developed, further enhancing student learning.

Problem statement For WebRTC to be effective in education, audio and video transmission must be of low latency and high quality for real-time interaction and collaboration, data transmission must be reliable to ensure a seamless learning experience, it must be compatible across devices and platforms to reach a broader audience, and educational tools and applications must be developed to enhance the learning experience. Protecting sensitive information and personal data is also essential for educational usage.

Literature to date WebRTC has been used in several studies and research projects for educational technology. Some of these studies have focused on using WebRTC for online classrooms, distance learning, and collaboration between students and teachers [92, 125–129]. A recent study noted that WebRTC offers the advantage of using limited internet connections due to its low bandwidth requirements [126]. In another study, Augmented reality (AR) was used to provide interactivity in the process [58]. Among the study's findings was that WebRTC provided a high level of service and could support real-time multimedia communication between students and teachers in real-time [127, 130]. Studies show that WebRTC can be used for educational technology, providing a flexible and effective way for students and teachers to collaborate and communicate in real-time. Despite this, additional research is required to fully understand the potential and limitations of WebRTC in the context of education. Similarly, videoconferencing for a mathematics class in which the whiteboard can be shown is considered [131].

Research gap WebRTC's potential for education needs to be realised by addressing several research gaps: For effective online collaboration and interaction between students and teachers, low latency is critical for reducing latency in real-time communication. Despite many studies about this, these studies are not mature because of the need for reliability, stability, security and privacy. Another research gap concerns the development of educational tools and applications that use WebRTC technology effectively. These include virtual and augmented reality simulations and collaborative online learning environments.

5 Open research directions and limitations

5.1 Open research directions

In addition to the addressed research gaps for each application, as mentioned previously, some general open research shall be considered for enhancing WebRTC. These challenges include:

- Improved Quality of Service (QoS): WebRTC currently relies on the underlying network conditions to determine the quality of the call. To improve QoS, researchers are developing algorithms like Adaptive Bitrate (ABR) and Forward Error Correction (FEC) [132–136]. ABR adjusts the video quality based on available bandwidth, ensuring smoother streaming [137]. FEC adds redundant data to help recover lost packets, improving call reliability. For example, the ABR algorithm can dynamically adjust video resolution to maintain a stable experience even if network conditions fluctuate.
- Network Adaptation: WebRTC is currently limited to working on a single network, but struggles with seamless operation across different networks such as 3G, 4G, and WiFi. Researchers are working on algorithms and protocols that can handle transitions between networks without interrupting the communication.
- Security and Privacy: As WebRTC becomes more widely adopted, there is a growing need to ensure that the technology is secure and that users' privacy is protected. Algorithms for end-to-end encryption, such as SRTP, are used to encrypt media streams. Additionally, secure key exchange mechanisms like Diffie-Hellman are implemented to

protect encryption keys. For example, SRTP encrypts audio and video streams, while Diffie-Hellman ensures that encryption keys are securely exchanged between users.

- Interoperability: WebRTC primarily works with other WebRTC clients, but making it compatible with other communication technologies like SIP and Extensible Messaging and Presence Protocol (XMPP) is a focus area. Researchers are developing bridging solutions and gateways that convert WebRTC signals to formats compatible with these protocols [138].
- Machine Learning: WebRTC is increasingly used in XR applications where machine learning algorithms can enhance the user experience [24]. For example, machine learning can be used to improve video quality through real-time image processing or to predict and reduce latency by analysing network patterns. Algorithms like Convolutional Neural Networks (CNNs) can enhance video clarity and reduce artifacts during streaming.
- WebRTC on IoT: WebRTC has the potential to provide a low-latency, secure, and reliable communication channel between IoT devices and servers. Researchers are exploring how WebRTC can be used to transmit real-time data from sensors and control devices efficiently. For instance, WebRTC could enable real-time monitoring and control of smart home devices with minimal delay.
- Edge Computing: With the increasing adoption of WebRTC, edge computing is becoming more important in reducing latency and bandwidth consumption. Edge computing involves processing data closer to the source rather than in a central data centre [139]. This can reduce latency and improve performance for WebRTC applications. For example, deploying WebRTC servers on edge nodes can decrease the time it takes for data to travel between users and servers.
- Adaptation of non-video data: To make WebRTC architecture capable of accepting non-video data, improving its efficiency and reliability is necessary. To facilitate the transfer of non-video data over WebRTC, modifying the underlying protocol and network architecture may be necessary. New tools and technologies may also be developed to facilitate this data transfer. To ensure seamless integration with existing data transfer solutions, it is also necessary to ensure compatibility with existing data transfer technologies and systems.

5.2 Study limitations

While this review provides a comprehensive overview of WebRTC's capabilities and applications, several limitations must be acknowledged. First, the scope of the review was constrained by the exclusion of non-English studies and publications that did not provide explicit explanations of WebRTC's use. This may have led to the omission of significant contributions from non-English-speaking research communities or studies addressing relevant applications outside the reviewed context. Furthermore, although this study analysed numerous applications of WebRTC, there remains a lack of large-scale, real-world deployment data to validate the findings of many reviewed papers, particularly in emerging areas such as virtual reality and telemedicine. Furthermore, while the systematic review methodology was robust, the rapidly evolving nature of WebRTC and its integration into various technologies means that the findings may quickly become outdated as new advancements emerge. Finally, the study primarily focused on the technical aspects of WebRTC and did not extensively explore the socio-economic or legal challenges surrounding its widespread adoption, which could provide important insights into its future development.

6 Conclusions

This review highlights its versatile capabilities and potential for real-time communication applications beyond traditional video conferencing. Through an extensive evaluation of 83 studies, it was demonstrated that WebRTC has established itself as a crucial tool across various sectors, including telemedicine, online gaming, education, and live streaming. The low-latency, peer-to-peer communication feature, combined with robust security protocols, provides unique advantages for high-quality, real-time data transmission. However, despite its adoption across numerous industries, several critical areas of improvement remain. This study identifies major research gaps, such as the need for improved scalability, enhanced security, and better handling of non-video data. Furthermore, WebRTC's integration with emerging technologies, like machine learning and edge computing, presents new opportunities for its advancement. Ultimately, the paper concludes that WebRTC is positioned to significantly influence future developments in real-time communication applications, but there is a need for focused research to address its current limitations and explore new applications.

List of Abbreviations WebRTC: Web Real-time Communication SIP: Session Initiation Protocol NATS: Network Address Translation Traversal ICE: Interactive Connectivity Establishment STUN: Session Traversal Utilities for NAT TURN: Traversal Using Relay around NAT SRTP: Secure Real-time Transport Protocol DTLS: Datagram Transport Layer Security UDP: User Datagram Protocol QoS: Quality of Service FEC: Forward Error Correction IoT: Internet of Things XR: Extended Reality VR: Virtual Reality AR: Augmented Reality ABR: Adaptive Bitrate P2P: Peer-to-Peer W3C: World Wide Web Consortium IETF: The Internet Engineering Task Force XMPP: Extensible Messaging and Presence Protocol CNN: Convolutional Neural Networks

Data Availability The data is available upon request.

References

1. Almousa O, Zhang R, Dimma M, Yao J, Allen Arden, Chen L, Heidari P, Qayumi K (2021) Virtual reality technology and remote digital application for tele-simulation and global medical education: An innovative hybrid system for clinical training. Simulation & Gaming 52(5):614–634
2. Radivojevic K, Clark N, Klempay A, Brenner P (2024) Defending novice user privacy: An evaluation of default web browser configurations. Compute Secur 140:103784
3. Kirchner R, Koch S, Kamangar N, Klein D, Johns M (2024) A black-box privacy analysis of messaging service providers' chat message processing. Proc Priv Enh Technol 3:1–18
4. Loreto S, Romano SP (2012) Real-time communications in the web: Issues, achievements, and ongoing standardization efforts. IEEE Int Comput 16(5):68–73
5. Li L, Zhang X (2012) Research on the integration of rtcweb technology with ip multimedia subsystem. In: 2012 5th International congress on image and signal processing, pp 1158–1161. IEEE
6. Antunes M, Silva C, Barranca J (2016) A telemedicine application using webrtc. Proc Comput Sci, 100:414–420. International Conference on ENTERprise Information Systems/International Conference

on Project MANagement/International Conference on Health and Social Care Information Systems and Technologies, CENTERIS/ProjMAN / HCist 2016
7. Loreto S, Romano SP (2017) How far are we from webrtc-1.0? an update on standards and a look at what's next. IEEE Commun Mag 55(7):200–207
8. Alvestrand H (2021) Rfc 8825: Overview: Real-time protocols for browser-based applications
9. Gouaillard A, Roux L (2017) Real-time communication testing evolution with webrtc 1.0. In: 2017 Principles, systems and applications of IP telecommunications (IPTComm), pp 1–8. IEEE
10. Garcia B, Gortazar F, Lopez-Fernandez L, Gallego M, Paris M (2017) Webrtc testing: challenges and practical solutions. IEEE Commun Stand Mag 1(2):36–42
11. Keele S, et al (2007) Guidelines for performing systematic literature reviews in software engineering. Technical report, Technical report, Ver. 2.3 EBSE Technical Report. EBSE
12. Velez G, Perez J, Martin A (2022) 5g mec-enabled vehicle discovery service for streaming-based cam applications. Multimed Tools Appl 81(9):12349–12370
13. Gupta I, Singh AK, Lee CN, Buyya R (2022) Secure data storage and sharing techniques for data protection in cloud environments: A systematic review, analysis, and future directions. IEEE Access 10:71247–71277
14. Diallo B, Ouamri A, Keche M (2023) A hybrid approach for webrtc video streaming on resource-constrained devices. Electronics 12(18):3775
15. Bentaleb A, Lim M, Akcay MN, Begen AC, Hammoudi S, Zimmermann R (2023) Toward one-second latency: Evolution of live media streaming. arXiv:2310.03256
16. Bambach J. Innovative web service for person authentication for electronic examination systems with webrtc. B.S. thesis, Technische Universität Darmstadt
17. Sergiienko A (2015) WebRTC Cookbook. Packt Publishing Ltd
18. Stolley K (2024) Programming WebRTC. Pragmatic Bookshelf
19. Naik AR, Keshavamurthy BN (2020) Next level peer-to-peer overlay networks under high churns: a survey. Peer-to-Peer Netw Appl 13(3):905–931
20. Zhang L, Li W, Yiyan W, Lafleche S, Hong Z, Park S, Lee J-Y, Kim H-M, Hur N, Iradier E et al (2020) Using layered division multiplexing for wireless in-band distribution links in next generation broadcast systems. IEEE Trans Broadcast 67(1):68–82
21. Reisinger T, Wagner I, Boiten EA (2022) Security and privacy in unified communication. ACM Comput Surv (CSUR) 55(3):1–36
22. Jansen B, Goodwin T, Gupta V, Kuipers F, Zussman G (2018) Performance evaluation of webrtc-based video conferencing. ACM SIGMETRICS Perform Eval Rev 45(3):56–68
23. Smirnov N, Tomforde S (2024) Real-time rate control of webrtc video streams in 5g networks: Improving quality of experience with deep reinforcement learning. J Syst Archit 148:103066
24. Sakakibara K, Ohzahata S, Yamamoto R (2024) Deep learning-based no-reference video streaming qoe estimation using webrtc statistics. In: 2024 International conference on artificial intelligence in information and communication (ICAIIC), pp 365–370. IEEE
25. Francisco N, Baumann O, Le Tanou J, Fliam R (2024) Ultra-low latency video delivery over webrtc data channels. In: Proceedings of the 3rd mile-high video conference, pp 88–89
26. Nakazato J, Nakagawa K, Itoh K, Fontugne R, Tsukada M, Esaki H (2024) Webrtc over 5 g: A study of remote collaboration qos in mobile environment. J Netw Syst Manag 32(1):1
27. Ran Y, Zhang T, Huang W, Xia S, Luo J (2023) isaw: Intelligent super-resolution-assisted adaptive webrtc video streaming. In: Proceedings of the 29th annual international conference on mobile computing and networking, pp 1–3
28. De Fré M, van der Hooft J, Wauters T, De Turck F. (2024) Scalable mdc-based volumetric video delivery for real-time one-to-many webrtc conferencing. In: Proceedings of the 15th ACM multimedia systems conference, pp 121–131
29. Al-Khayyat AT, Mahmood SA (2023) Peer-to-peer media streaming with html5. Int J Electr Comput Eng 13(2):2356
30. Sacchetto M, Gastaldi P, Chafe C, Rottondi C, Servetti A (2022) Web-based networked music performances via webrtc: a low-latency pcm audio solution. J Aud Eng Soc 70(11):926–937
31. Sassu A, Saenz-Cogollo JF, Agelli M (2021) Deep-framework: A distributed, scalable, and edge-oriented framework for real-time analysis of video streams. Sensors 21(12):4045
32. Huang CY, Cheng YC, Huang GZ, Fan CL, Hsu CH (2021) On the performance comparisons of native and clientless real-time screen-sharing technologies. ACM Trans Multimed Comput Commun Appl (TOMM) 17(2):1–26
33. Fowdur TP, Ramkorun N, Chiniah PK (2020) Performance analysis of webrtc and sip-based audio and video communication systems. SN Comput Sci 1:1–22

34. Amirante A, Castaldi T, Miniero L, Romano SP (2013) On the seamless interaction between webrtc browsers and sip-based conferencing systems. IEEE Commun Mag 51(4):42–47
35. Edan N, Abdullah EY (2020) Design and implementation of a novel secured and wide webrtc signalling mechanism for multimedia over internet. Int J Electr Comput Eng 10(5):5430
36. Toğay C (2020) A practical key agreement scheme for videoconferencing. Multimed Tools Appl 79(33–34):23711–23728
37. Fakis A, Karopoulos G, Kambourakis G (2020) Neither denied nor exposed: Fixing webrtc privacy leaks. Future Int 12(5):92
38. García B, Gortázar F, Gallego M, Hines A (2020) Assessment of qoe for video and audio in webrtc applications using full-reference models. Electronics 9(3):462
39. Tang D, Zhang L (2020) Audio and video mixing method to enhance webrtc. IEEE Access 8:67228–67241
40. Palacios S, Santos V, Barsallo E, Bhargava B (2019) Miostream: a peer-to-peer distributed live media streaming on the edge. Multimed Tools Appl 78:24657–24680
41. Petrangeli S, Pauwels D, Van Der Hooft J, Žiak M, Slowack J, Wauters T, De Turck F (2019) A scalable webrtc-based framework for remote video collaboration applications. Multimed Tools Appl 78:7419–7452
42. Carlucci G, De Cicco L, Holmer S, Mascolo S (2017) Congestion control for web real-time communication. IEEE/ACM Trans Netw 25(5):2629–2642
43. Teykhrib AP (2015) Prospects in application of the technology for of reception and transmission of streaming data in web browser. J Theor Appl Inf Technol 80(1):84
44. Barnes RL, Thomson M (2014) Browser-to-browser security assurances for webrtc. IEEE Int Comput 18(6):11–17
45. Lopez-Fernandez L, Gallego M, García B, Fernández-López D, Lopez FJ (2014) Authentication, authorization, and accounting in webrtc paas infrastructures: The case of kurento. IEEE Int Comput 18(6):34–40
46. Alahmadi M, Pocta P, Melvin H (2021) An adaptive bitrate switching algorithm for speech applications in context of webrtc. ACM Trans Multimed Comput Commun Appl (TOMM) 17(4):1–21
47. Gortázar F, Gallego M, Maes-Bermejo M, Chicano-Capelo I, Santos C (2022) Cost-effective load testing of webrtc applications. J Syst Softw 193:111439
48. Blum N, Lachapelle S, Alvestrand H (2021) Webrtc: Real-time communication for the open web platform. Commun ACM 64(8):50–54
49. Rhinow F, Veloso PP, Puyelo C, Barrett S, Nuallain EO (2014) P2p live video streaming in webrtc. In: 2014 World Congress on Computer Applications and Information Systems (WCCAIS), pp 1–6
50. Nurminen JK, Meyn AJ, Jalonen E, Raivio Y, Marrero RG (2013) P2p media streaming with html5 and webrtc. In: 2013 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp 63–64
51. Brambilla G, Amoretti M, Zanichelli F (2018) An open web application framework for peer-to-peer location-based services. Concurr Comput Pract Exp 30(20):v
52. de Paiva GM, Dias DR, Mota JH, Gnecco BB, Durelli VH, Trevelin LC (2018) Immersive and interactive virtual reality applications based on 3d web browsers. Multimed Tools Appl 77:347–361
53. Roy SG, Kanjilal U (2021) Web-based augmented reality for information delivery services: A performance study. desidoc. DESIDOC J Library Inf Technol 41(1):1
54. Fang D, Huahu X, Yang X, Bian M (2020) An augmented reality-based method for remote collaborative real-time assistance: from a system perspective. Mob Netw Appl 25:412–425
55. Jikadra R, Shah H, Kulkarni P (2019) Video calling with augmented reality using webrtc api. In: 2019 6th International conference on computing for sustainable global development (INDIACom), pp 363–367. IEEE (INDIACom), pp 363–367. IEEE
56. García B, Gallego M, Gortázar F, Bertolino A (2019) Understanding and estimating quality of experience in webrtc applications. Computing 101:1585–1607
57. Gunkel S, Prins M, Stokking H, Niamut O (2017) WebVR meets WebRTC: Towards 360-degree social VR experiences. IEEE
58. Maraševic J, Gavrovska A (2020) Virtual reality and webrtc implementation for web educational application development. In: 2020 28th Telecommunications forum (TELFOR), pp 1–4. IEEE
59. Muralikrishnan S (2021) A comparative study on cloud gaming performance using traditional, containers in fog nodes, and edge-enabled shared gpu architectures. Master's thesis, Dublin, National College of Ireland
60. Oros BI, Bâcu VI (2020) Renderlink remote rendering platform for computer games : A webrtc solution for streaming computer games. In: 2020 IEEE 16th International conference on intelligent computer communication and processing (ICCP), pp 555–561

61. Alhilal A, Braud T, Han B, Hui P (2022) Nebula: Reliable low-latency video transmission for mobile cloud gaming. In: Proceedings of the ACM web conference 2022, WWW '22, page 3407-3417, New York, NY, USA. Association for Computing Machinery
62. Lepola K (2020) Managing network delay for browser multiplayer games. Master's Thesis
63. Dobreff G, Frey D, Báder A, Pašić (2024) Empowering isps with cloud gaming user experience modeling: A nvidia geforce now use-case. In: Proceedings of the 27th conference on innovation in clouds, internet and networks (ICIN), pp 202–209. IEEE
64. Duan Q, Liang Z (2016) File sharing strategy based on webrtc. In: 2016 13th Web information systems and applications conference (WISA), pp 3–6
65. Dukiya R, Sharma S, Koppu S, Rao M (2017) Reliable file sharing in distributed operating system using web rtc. In: 2017 International conference on microelectronic devices, circuits and systems (ICMDCS), pp 1–4
66. Vashishth S, Sinha Y, Babu KH (2016) Addressing challenges in browser based p2p content sharing framework using webrtc. In: 2016 IEEE 30th international conference on advanced information networking and applications (AINA), pp 850–857
67. Ivan D, Mislav G, Gordan G (2020) Exploring webrtc potential for dicom file sharing. J Digit Imaging 33:697–707
68. García B, Kloos CD, Alario-Hoyos C et al (2022) Selenium-jupiter: A junit 5 extension for selenium webdriver. J Syst Softw 189:111298
69. Serreli L, Bingol G, Porcu S, et al (2023) Robust quic-based signaling for webrtc in impaired network conditions. In: 2023 IEEE International mediterranean conference on communications. IEEE
70. Little J, Schmeltz A, Cooper M, Waldrop T, Yarvis JS, Pruitt L, Dondanville K (2021) Preserving Continuity of Behavioral Health Clinical Care to Patients Using Mobile Devices. Mil Med 186(Supplement1):137–14, 101
71. Drnasin I, Grgic M, Gledec G (2020) Exploring webrtc potential for dicom file sharing. J Digit Imaging 33(3):697–707
72. Ha VK, Chai R, Nguyen HT (2020) A telepresence wheelchair with 360-degree vision using webrtc. Appl Sci 10(1):369
73. Patel V, Li CH, Rye V, Liu CS, Lerner A, Acharya J, Rajamohan AG (2022) A comparison of webrtc and conventional videoconferencing for synchronized remote medical image presentation. J Digit Imaging 35(1):68–76
74. Jang-Jaccard J, Nepal S, Celler B, Yan B (2016) Webrtc-based video conferencing service for telehealth. Computing 98(1–2):169–193
75. Kagathara D, Bhatt N (2019) Audio and video streaming in telepresence application using webrtc for healthcare system. In: Proceedings of the international conference on ISMAC in computational vision and bio-engineering 2018 (ISMAC-CVB), pp 1763–1772. Springer
76. Sime AW (2016) Webrtc: delivering telehealth in the browser. Mhealth 2
77. Pierleoni P, Pernini L, Palma L, Belli A, Valenti S, Maurizi L, Sabbatini L, Marroni A (2016) An innovative webrtc solution for e-health services. In: 2016 IEEE 18th international conference on e-health networking, applications and services (Healthcom), pp 1–6. IEEE
78. El Jaouhari S, Bouabdallah A, Bonnin JM (2018) A secure webrtc/wot-based health-care architecture enhanced with access control. In: 2018 International conference on information networking (ICOIN), pp 182–187. IEEE
79. El Jaouhari S, Bouabdallah A, Bonnin JM, Lemlouma T (2017) Toward a smart health-care architecture using webrtc and wot. In: Recent Advances in Information Systems and Technologies: Volume 3, pp 531–540. Springer
80. Voruganti T, Grunfeld E, Makuwaza T, Bender JL (2017) Web-based tools for text-based patient-provider communication in chronic conditions: scoping review. J Med Int Res 19(10):e366
81. Antunes M, Silva C, Barranca J (2016) A telemedicine application using webrtc. Proc Comput Sci 100:414–420
82. El Jaouhari S, Bouabdallah A (2018) A privacy safeguard framework for a webrtc/wot-based healthcare architecture. In: 2018 IEEE 42nd annual computer software and applications conference (COMPSAC), vol 2, pp 468–473. IEEE
83. Abdullah AS, Manoj A, Kishore GT, Selvakumar S (2021) A new approach to remote health monitoring using augmented reality with webrtc and webxr. In: 2021 22nd International arab conference on information technology (ACIT), pp 1–5. IEEE
84. Šarić M, Russo M, Kraljević L, Meter D (2024) Extended reality telemedicine collaboration system using webrtc. Sensors 24(1):27
85. Toğay C (2014) Webrtc technology for mobile devices. In: 2014 22nd Signal processing and communications applications conference (SIU), pp 256–259

86. Di Mauro M, Longo M (2015) Revealing encrypted webrtc traffic via machine learning tools. In: 2015 12th International joint conference on e-business and telecommunications (ICETE), vol 04, pp 259–266
87. Nasoz F, Shrestha C (2017) A web-based user interface for machine learning analysis. In: Yamamoto S (Ed) Human interface and the management of information: supporting learning, decision-making and collaboration, pp 444–453, Cham. Springer International Publishing
88. Chodorek A, Chodorek RR, Sitek P (2021) Uav-based and webrtc-based open universal framework to monitor urban and industrial areas. Sensors, 21(12)
89. Barradas D, Santos N (2020) Towards a scalable censorship-resistant overlay network based on webrtc covert channels. In: Proceedings of the 1st international workshop on distributed infrastructure for common good, pp 37–42
90. Barradas D, Santos N, Rodrigues L, Nunes V (2020) Poking a hole in the wall: Efficient censorship-resistant internet communications by parasitizing on webrtc. In: Proceedings of the 2020 ACM SIGSAC conference on computer and communications security, pp 35–48
91. SEYED C, BILONG JR, SIDI MA, NANNE MF (2022) Techno-pedagogical solution to support the improvement of the quality of education in technical and vocational training in mauritania. Int J Adv Comput Sci Appl 13(7)
92. Li G, Wang R, Zhou Q (2021) Design and application of college online education platform based on webrtc. Scalable Comput Pract Exp 22(2):139–148
93. Mutijarsa K, Bandung Y, Subekti LB (2017) Videoconference system for rural education: Issues, challenges, and solutions a title is fewest possible words. TELKOMNIKA (Telecomm Comput Electr Control) 15(4):1901–1911
94. Osipov IV, Volinsky AA, Nikulchev E, Prasikova AY (2016) Communication and gamification in the web-based foreign language educational system: Web-based foreign language educational system. Int J Web-Based Lear Teach Technol (IJWLTT) 11(4):22–34
95. Huang Y, Qiao X, Lai W, Dustdar S, Zhang J, Li J (2021) Enabling dnn acceleration with data and model parallelization over ubiquitous end devices. IEEE Int Things J 9(16):15053–15065
96. Gupta M, Thakur N, Bansal D, Chaudhary G, Davaasambuu B, Hua Q, et al (2022) Cnn-lstm hybrid real-time iot-based cognitive approaches for islr with webrtc: auditory impaired assistive technology. J Healthcare Eng 2022
97. Wang B, Guan S, Wang N (2022) Research on the application of voice interaction technology in smart water supply systems. In: ACM International conference proceeding series
98. Izima O, de Fréin R, Malik A (2021) A survey of machine learning techniques for video quality prediction from quality of delivery metrics. Electronics 10(22)
99. Ammar D, De Moor K, Skorin-Kapov L, Fiedler M, Heegaard PE (2019) Exploring the usefulness of machine learning in the context of webrtc performance estimation. In: 2019 IEEE 44th conference on local computer networks (LCN), pp 406–413
100. Yan S, Guo Y, Chen Y, Xie F (2019) Predicting freezing of webrtc videos in wifi networks. In: Zheng J, Xiang W, Lorenz P, Mao S, Yan F (Eds) Ad Hoc Networks, pp 292–301, Cham. Springer International Publishing
101. Qiao X, Ren P, Dustdar S, Liu L, Ma H, Chen J (2019) Web ar: A promising future for mobile augmented reality-state of the art, challenges, and insights. Proc IEEE 107(4):651–666
102. Kotevski Z, Nedelkovski I, Markoski A, Veljanovska K, Bocevska A, Dimovski T, Rendevski N. (2022) Media streaming technologies: Current and future trends. In: International conference on applied internet and information technologies AIIT
103. Cruzat La Rosa A (2024) Re-encoding resistance: Towards robust covert channels over webrtc video streaming. Master's thesis, University of Waterloo
104. Sun Z, Shmatikov V (2024) Differential degradation vulnerabilities in censorship circumvention systems. arXiv:2409.06247
105. Bingöl G, Porcu S, Floris A, Atzori L (2024) Qoe estimation of webrtc-based audio-visual conversations from facial and speech features. ACM Trans Multimed Comput Commun Appl 20(5):1–23
106. WebRTC World (n.d.) How webrtc is transforming the online casino gaming industry.
107. Nazerevich D (2024) What is webrtc?
108. Dobreff G, Frey D, Báder A, Pašić A (2024) Empowering isps with cloud gaming user experience modeling: A nvidia geforce now use-case. In: 2024 27th Conference on Innovation in Clouds, Internet and Networks (ICIN), pp 202–209. IEEE
109. Zhang Y, Cheng S, Guo Z, Zhang X (2024) Inferring video streaming quality of real-time communication inside network. IEEE Trans Circ Syst Vid Technol
110. Duan Q, Liang Z (2016) File sharing strategy based on webrtc. In: 2016 13th Web information systems and applications conference (WISA), pp 3–6. IEEE

111. Grigorik I (2013) High Performance Browser Networking: What every web developer should know about networking and web performance. "O'Reilly Media, Inc."
112. Yang H, Kim D, Lee D, Park S (2024) Blockchain-based reliable and secure file access control for distributed public node. In: 2024 IEEE international conference on consumer electronics (ICCE), pp 1–6. IEEE
113. Usón J, Muñoz V, Cortés C, Berjón D, Morán F, Díaz C, Gutierrez J, Jaureguizar F, García N, Cabrera J (2024) Real-time free viewpoint video for immersive videoconferencing. In: 2024 16th International conference on quality of multimedia experience (QoMEX), pp 171–174. IEEE
114. Tricahyadinata I, Za SZ (2017) An analysis on the use of google adwords to increase e-commerce sales. SZ Za and I. Tricahyadinata (2017) Int J Soc Sc Manage 4:60–67
115. Zhang L, Zhou F, Mislove A, Sundaram R (2013) Maygh: Building a cdn from client web browsers. In: Proceedings of the 8th ACM european conference on computer systems, pp 281–294
116. Kumar N (2018) E-commerce in india: an analysis of present status, challenges and opportunities. Int J Manag Stud 5(2):3
117. Anil K (2019) Internet and wold wide web (paper code: Mm-409/ib-419). ddeg
118. Baretto A, Pudussery N, Subramaniam V, Siddiqui A (2021) Real-time webrtc based mobile surveillance system. Int J Eng Manag Res 11(3):30–35
119. Osman A, Abozariba R, Asyhari AT, Aneiba A, Farah MB (2022) Detection of javascript injection eavesdropping on webrtc communications. In: 2022 IEEE 23rd international symposium on a world of wireless, mobile and multimedia networks (WoWMoM), pp 541–547. IEEE
120. Iiyoshi K, Gebremedhin R, Gokhale V, Eid M (2021) Plug-and-play haptic interaction for tactile internet based on webrtc. In: Intelligent technologies for interactive entertainment: 12th EAI international conference, INTETAIN 2020, Virtual Event, December 12-14, 2020, Proceedings, pp 65–81. Springer
121. Kurillo G, Yang AY, Shia V, Bair A, Bajcsy R (2016) New emergency medicine paradigm via augmented telemedicine. In Virtual, augmented and mixed reality: 8th international conference, VAMR 2016, Held as Part of HCI International 2016, Toronto, Canada, July 17-22, 2016. Proceedings 8, pp 502–511. Springer
122. Chodorek A, Chodorek RR, Yastrebov A (2021) Weather sensing in an urban environment with the use of a uav and webrtc-based platform: A pilot study. Sensors 21(21):7113
123. Chodorek A, Chodorek RR, Sitek P (2021) Uav-based and webrtc-based open universal framework to monitor urban and industrial areas. Sensors 21(12):4061
124. Chodorek A, Chodorek RR, Yastrebov A (2022) The prototype monitoring system for pollution sensing and online visualization with the use of a uav and a webrtc-based platform. Sensors 22(4):1578
125. Petrangeli S, Pauwels D, van der Hooft J, Slowack J, Wauters T, De Turck F (2018) Dynamic video bitrate adaptation for webrtc-based remote teaching applications. In NOMS 2018-2018 IEEE/IFIP network operations and management symposium, pp 1–5. IEEE
126. Ouya S, Seyed C, Mbacke AB, Mendy G, Niang I (2015) Webrtc platform proposition as a support to the educational system of universities in a limited internet connection context. In: 2015 5th World congress on information and communication technologies (WICT), pp 47–52. IEEE
127. Ouya S, Sylla K, Faye PM, Sow MY, Lishou C (2015) Impact of integrating webrtc in universities'e-learning platforms. In: 2015 5th World congress on information and communication technologies (WICT), pp 13–17. IEEE
128. Liao Y, Wang Z, Luo Y (2016) The design and implementation of a webrtc based online video teaching system. In: 2016 IEEE advanced information management, communicates, electronic and automation control conference (IMCEC), pp 137–140. IEEE
129. Li X, Yan Y, Ma Y, Zhao M, Dan (2021) A solution of webrtc-based interactive smart classroom for college teaching. In: 2021 IEEE 3rd international conference on computer science and educational informatization (CSEI), pp 22–26. IEEE
130. Zhang J, Zhang Y, Qi B, Zhao H, Watanabe T (2019) A webrtc e-learning system based on kurento media server. In: E-learning and games: 12th international conference, edutainment 2018, Xi'an, China, June 28–30, 2018, Proceedings 12, pp 331–335. Springer
131. Faye PM, Gueye AD, Lishou C (2018) Virtual classroom solution with webrtc in a collaborative context in mathematics learning situation. In: Innovation and interdisciplinary solutions for underserved areas: first international conference, InterSol 2017 and Sixth Collogue National sur la Recherche en Informatique et ses Applications, CNRIA 2017, Dakar, Senegal, April 11–12, 2017, Proceedings 1, pp 66–77. Springer
132. Azeem MM, Abozariba R, Asyhari AT (2023) Exploiting short block and concatenated codes for reliable communications within the coexistence of 5g-nr-u and wifi. IEEE Trans Veh Technol 72(2):1893–1908
133. Azeem MM, Khan AB, Azeem U (2015) On the performance of tanner graph based and viterbi decoding for erasure recovery. In: 2015 IEEE 82nd vehicular technology conference (VTC2015-Fall), pp 1–5

134. Azeem MM, Khan AB, Azeem U (2017) Application of short erasure correcting codes for cognitive radio. In: 2017 IEEE 86th vehicular technology conference (VTC-Fall), pp 1–5
135. Azeem MM, Tortelier P, Le Ruyet D (2012) Single parity check product codes for erasure recovery in opportunistic spectrum access. In: 2012 International symposium on wireless communication systems (ISWCS), pp 76–80
136. Tortelier P, Azeem MM (2011) Improving the erasure recovery performance of short codes for opportunistic spectrum access. In: 2011 The 14th international symposium on wireless personal multimedia communications (WPMC), pp 1–5
137. Ahmad A, Tristan B, Bo H, Pan H (2022) Nebula: Reliable low-latency video transmission for mobile cloud gaming. In: Proceedings of the ACM web conference, vol 2022, pp 3407–3417
138. Shreya GP, Pradhyumna P, et al (2021) Internetworking gateway between webrtc to sip to integrate real-time audio video communication. In: 2021 Third international conference on inventive research in computing applications (ICIRCA), pp 1480–1485. IEEE
139. Mtowe DP, Kim DM (2023) Edge-computing-enabled low-latency communication for a wireless networked control system. Electronics 12(14):3181